Amendments to the Claims

Claim 1 (original): A computer program product embodied on one or more computer-readable media, the computer program product adapted for efficiently transforming extensible structured documents and comprising:

computer-readable program code means for identifying a source document type;

computer-readable program code means for specifying one or more fast transformations to be performed on documents of the source document type;

computer-readable program code means for specifying a source node description and a target node description for each of the specified fast transformations;

computer-readable program code means for storing transformation information for each of the specified fast transformations, the transformation information comprising a transformation identifier, the source node description, and the target node description; and

computer-readable program code means for processing incoming source documents to generate output documents using the stored transformation information, further comprising:

computer-readable program code means for receiving a source document;

computer-readable program code means for selecting, manually or based upon a comparison of the received source document to the stored transformation information, zero or more fast transformations to be performed;

computer-readable program code means for applying the selected fast transformations; and

computer-readable program code means for generating one or more output documents using a result of the computer-readable program code means for applying.

Claim 2 (original): The computer program product according to Claim 1, wherein the received source document is an Extensible Markup Language (XML) document.

Claim 3 (original): The computer program product according to Claim 2, further comprising computer-readable program code means for parsing the XML document.

Claim 4 (original): The computer program product according to Claim 1, wherein the received source document is an array-based representation of an Extensible Markup Language (XML) document.

Claim 5 (original): The computer program product according to Claim 4, and wherein the computer-readable program code means for applying the selected transformations further comprises computer-readable program code means for manipulating selected nodes by manipulating the array-based representation.

Claim 6 (original): The computer program product according to Claim 1, wherein the received source document is a machine-oriented markup language document.

Claim 7 (original): The computer program product according to Claim 1, wherein the received source document is an array-based representation of a machine-oriented markup language document.

Claim 8 (original): The computer program product according to Claim 1, wherein the received source document is a parsed representation of an extensible document.

Claim 9 (original): The computer program product according to Claim 1, wherein the source node description identifies one or more source nodes in an input document of the source document type and wherein the target node description identifies zero or more target nodes in an output tree to be generated in the one or more output documents.

Claim 10 (currently amended): The computer program product according to Claim 1, wherein the computer-readable program code means for processing further comprises computer-readable program code means for applying a general purpose transformation engine to perform transformations other than the selected fast transformations, wherein the general purpose transformation engine is a stylesheet engine, and wherein the computer-readable program code means for generating uses the result of the computer-readable program code means for applying the selected fast transformations as well as a result of the computer-readable program code means for applying the general purpose transformation engine.

Claim 11 (original): The computer program product according to Claim 10, wherein the stylesheet engine is an Extensible Stylesheet Language (XSL) engine.

Claim 12 (currently amended): A system for ~~efficiently~~ transforming extensible structured documents, comprising:

means for specifying fast transformations to be applied to incoming source documents, further comprising:

means for specifying a signature that identifies a source document type;

means for specifying one or more fast transformations to be performed on documents that match the specified signature;

means for specifying a source node description and a target node description for each of the specified fast transformations; and

means for storing transformation information for each of the specified fast transformations, the transformation information comprising a transformation identifier, the source node description, and the target node description;

means for applying the fast transformations to particular incoming source documents matching criteria of the specified fast transformations; and

means for applying general purpose transformations to incoming source documents not matching criteria of the specified fast transformations.

Claim 13 (canceled)

Claim 14 (currently amended): The system according to Claim [[13]] 12, wherein the means for applying the fast transformations further comprises:

means for receiving a source document;

means for selecting, manually or based upon a comparison of the received source document to the stored transformation information, zero or more fast transformations to be performed; and

means for applying the selected fast transformations by manipulating selected nodes of the received source document according to the selected fast transformations.

Claim 15 (currently amended): The system according to Claim [[12]] 14, wherein the received source document is an Extensible Markup Language (XML) document.

Claim 16 (original): The system according to Claim 15, further comprising means for parsing the XML document.

Claim 17 (currently amended): The system according to Claim [[12]] 14, wherein the received source document is an array-based representation of an Extensible Markup Language (XML) document.

Claim 18 (original): The system according to Claim 14, wherein the received source document is an array-based representation of an Extensible Markup Language (XML) document, and wherein the means for applying the selected fast transformations by manipulating selected nodes further comprises means for manipulating the array-based representation.

Claim 19 (currently amended): The system according to Claim [[12]] 14, wherein the received source document is a machine-oriented markup language document.

Claim 20 (currently amended): The system according to Claim [[12]] 14, wherein the received source document is an array-based representation of a machine-oriented markup language document.

Claim 21 (currently amended): The system according to Claim [[12]] 14, wherein the received source document is a parsed representation of an extensible document.

Claim 22 (currently amended): The system according to Claim 12, wherein the source node description identifies one or more source nodes in an input document of the source document type and wherein the target node description identifies zero or more target nodes ~~in an output tree~~ to be generated in [[the]] one or more output documents, responsive to operation of the means for applying the fast transformations.

Claim 23 (currently amended): The system according to Claim 12, wherein the general purpose ~~transformation engine is~~ transformations are applied using a stylesheet engine.

Claim 24 (original): The system according to Claim 23, wherein the stylesheet engine is an Extensible Stylesheet Language (XSL) engine.

Claim 25 (currently amended): A method for ~~efficiently~~ transforming extensible structured documents, comprising the steps of:

specifying fast transformations to be applied to incoming source documents, further comprising the steps of:

specifying a signature that identifies a source document type;

specifying one or more fast transformations to be performed on documents that match the specified signature;

specifying a source node description and a target node description for each of the specified fast transformations; and

storing transformation information for each of the specified fast transformations, the transformation information comprising a transformation identifier, the source node description, and the target node description;

applying the fast transformations to particular incoming source documents matching criteria of the specified fast transformations; and

applying general purpose transformations to incoming source documents not matching criteria of the specified fast transformations.

Claim 26 (canceled)

Claim 27 (currently amended): The method according to Claim [[26]] 25, wherein the step of applying the fast transformations further comprises the steps of:

receiving a source document;

selecting, manually or based upon a comparison of the received source document to the stored transformation information, zero or more fast transformations to be performed; and applying the selected fast transformations by manipulating selected nodes of the received source document according to the selected fast transformations.

Claim 28 (currently amended): The method according to Claim [[25]] 27, wherein the received source document is an Extensible Markup Language (XML) document.

Claim 29 (original): The method according to Claim 28, further comprising the step of parsing the XML document.

Claim 30 (currently amended): The method according to Claim [[25]] 27, wherein the received source document is an array-based representation of an Extensible Markup Language (XML) document.

Claim 31 (original): The method according to Claim 27, wherein the received source document is an array-based representation of an Extensible Markup Language (XML) document, and wherein the step of applying the selected fast transformations by manipulating selected nodes further comprises the step of manipulating the array-based representation.

Claim 32 (currently amended): The method according to Claim [[25]] 27, wherein the received source document is a machine-oriented markup language document.

Claim 33 (currently amended): The method according to Claim [[25]] 27, wherein the received source document is an array-based representation of a machine-oriented markup language document.

Claim 34 (currently amended): The method according to Claim [[25]] 27, wherein the received source document is a parsed representation of an extensible document.

Claim 35 (currently amended): The method according to Claim 25, wherein the source node description identifies one or more source nodes in an input document of the source document type and wherein the target node description identifies zero or more target nodes ~~in an output tree~~ to be generated in [[the]] one or more output documents, responsive to operation of the step of applying the fast transformations.

Claim 36 (currently amended): The method according to Claim 25, wherein the general purpose ~~transformation engine is~~ transformations are applied using a stylesheet engine.

Claim 37 (original): The method according to Claim 36, wherein the stylesheet engine is an Extensible Stylesheet Language (XSL) engine.

Claim 38 (original): The method according to Claim 27, further comprising the step of preloading one or more templates prior to operation of the step of applying the selected fast transformations.

Claim 39 (original): The method according to Claim 25, further comprising using a result of the step of applying the fast transformations and a result of the step of applying general purpose transformations to create an output document, and wherein the source document and/or the output document may be represented as in-memory structures which may have been produced by or may be sent to another software process.